Quantum Information and Computation, Vol. 8, No. 1&2 (2008) 0012–0029

QUANTUM ALGORITHM DESIGN USING DYNAMIC LEARNING

E.C. BEHRMAN
Department of Physics, Wichita State University
Wichita, KS 67260-0032,USA

J.E. STECK
Department of Aerospace Engineering, Wichita State University
Wichita, KS 67260-0044, USA

P. KUMAR
Department of Electrical and Computer Engineering, Wichita State University
Wichita, KS 67260-0044, USA

K.A. WALSH
Department of Physics, Wichita State University
Wichita, KS 67260-0032, USA

Received 15 October 2006
Revised 7 August 2007

We present a dynamic learning paradigm for "programming" a general quantum computer. A learning algorithm is used to find the control parameters for a coupled qubit system, such that the system at an initial time evolves to a state in which a given measurement corresponds to the desired operation. This can be thought of as a quantum neural network. We first apply the method to a system of two coupled superconducting quantum interference devices (SQUIDs), and demonstrate learning of both the classical gates XOR and XNOR. Training of the phase produces a gate congruent to the CNOT modulo a phase shift. Striking out for somewhat more interesting territory, we attempt learning of an entanglement witness for a two qubit system. Simulation shows a reasonably successful mapping of the entanglement at the initial time onto the correlation function at the final time for both pure and mixed states. For pure states this mapping requires knowledge of the phase relation between the two parts; however, given that knowledge, this method can be used to measure the entanglement of an otherwise unknown state. The method is easily extended to multiple qubits or to quNits.

Keywords: quantum algorithm, entanglement, dynamic learning

Communicated by: S Braunstein & B Terhal

1 Introduction

Recently there has been growing interest in quantum computing [1, 2]. The possibilities seem vast. Beyond the improvements in size and speed, is the ability, at least in principle, to do classically impossible calculations. Two aspects of quantum computing make this possible: quantum parallelism and entanglement. While a computational setup can be constructed [3] which makes use of quantum parallelism (superposition) only, use and manipulation of

entanglement as well realizes the full power of quantum computing and communication [4, 5, 6].

The major bottleneck to the use of quantum computers, once they are designed and built, is the paucity of algorithms that can make use of their power. At present, there are only a few major algorithms: Shor's factorization [7], Grover's data base search [8], the Jones polynomial approximation [9]. It is not yet at all clear that a way will be or can be found to generate algorithms efficiently to solve general problems on quantum computers, as pointed out by Nielsen[10], though some recent work[10, 12] using a geometric approach may prove fruitful.

In previous work [3], we have proposed the use of quantum adaptive computers to answer this need. An adaptive computer, since it can be trained, adapts to learn and in a sense constructs its own algorithm for the problem from the training set supplied. A quantum neural computer shows promise for constructing algorithms to solve problems that are inherently quantum mechanical. Here, we develop a dynamic learning algorithm for training a quantum computer and demonstrate successful learning of some simple benchmark applications. In addition, we show that this method can be used for learning of an entanglement witness[13] for an input state. We show that our witness approximately reproduces the entanglement of formation for large classes of states. Generalization to systems of more than two qubits [14], or to multiple level systems [15], is straightforward.

2 Coupled Two-qubit System: QNN

Two interacting qubits, labeled A and B, can be used to build a quantum gate where each qubit interacts with a coupling (connectivity) that can be externally adjusted. This is a dynamical system that can be prepared in an initial (input) state, which then evolves in time to the point where it can be measured at some final time to yield an output. Adjustable physical parameters of the qubits allow "programming" to "compute" a specified output in response to a given input.

Consider a two-qubit quantum system that evolves in time according to the Hamiltonian:

$$H = K_A\sigma_{xA} + K_B\sigma_{xB} + \varepsilon_A\sigma_{zA} + \varepsilon_B\sigma_{zB} + \zeta\sigma_{zA}\sigma_{zB} \tag{1}$$

where $\{\sigma\}$ are the Pauli operators corresponding to each of the two qubits, A and B, K_A and K_B are the tunneling amplitudes, ε_A and, ε_B are the biases, and ζ the qubit-qubit coupling. This Hamiltonian can also be written [16] as

$$\begin{aligned} H = \quad \textstyle\sum_{n_1,n_2=0,1} E_{n_1,n_2}|n_1,n_2\rangle\langle n_1,n_2| \quad & -\frac{E_{J1}}{2}\sum_{n_2=0,1}(|0\rangle\langle 1| + |1\rangle\langle 0|)\otimes|n_2\rangle\langle n_2| \\ & -\frac{E_{J2}}{2}\sum_{n_1=0,1}|n_1\rangle\langle n_1|\otimes(|0\rangle\langle 1| + |1\rangle\langle 0|) \end{aligned} \tag{2}$$

using the two-qubit charge basis $|00\rangle, |10\rangle, |01\rangle, |11\rangle$. This could represent a number of different possible physical systems, *e.g.*, trapped ions [17] or nuclear magnetic resonance [18]. Here we take as our physical model an electrostatically coupled two-SQUID system [16, 19]. With appropriately timed pulses of the bias amplitudes ε_A and ε_B , entangled states can be prepared, and logic gates such as the CNOT or Toffoli reproduced [16, 20]. While we consider here a SQUID system with the particular physical identifications, above, of the parameters in the Hamiltonian, the method is by no means limited to that physical implementation of a

two-quibit system, or, indeed, to a two-qubit system. It is easy to see how to generalize to any Hamiltonian containing appropriately adjustable parameters.

The density matrix of the system as a function of time obeys the Schrdinger equation $\frac{d\rho}{dt} = \frac{1}{i\hbar}[H, \rho]$, whose formal solution is $\rho(t) = \exp(iLt)\rho(t)$. This is of similar mathematical form to the equation for information propagation in a neural network, as follows. For a traditional artificial neural network, the calculated activation ϕ_i of the i^{th} neuron is performed on the signals $\{\phi_i\}$ from the other neurons in the network, and is given by $\phi_i = \sum_j w_{ij} f_j(\phi_j)$, where w_{ij} is the weight of the connection from the output of neuron j to neuron i, and f_i is a bounded differentiable real valued neuron activation function for neuron j. Individual neurons are connected together into a network to process information from a set of input neurons to a set of output neurons. The network is an operator F_W on the input vector ϕ_{input}. F_W depends on the neuron connectivity weight matrix W and propagates the information forward to calculate an output vector ϕ_{output}; that is, $\phi_{output} = F_W \phi_{input}$. The time evolution equation for the density matrix maps the initial state (input) to the final state (output) in much the same way. The parameters playing the role of the adjustable weights in the neural network are the set $\{K_A, K_B, \varepsilon_A, \varepsilon_B, \zeta\}$, all of which can be adjusted experimentally as functions of time for the SQUID system under consideration [16, 19]. By adjusting the parameters using a neural network type learning algorithm we can train the system to evolve in time to a final state at the final time t_f for which the desired measure has been mapped to the function we wish the net to compute: logic gates, or, since the time evolution is quantum mechanical, a quantum function like the entanglement. Indeed, if we think of the time evolution operator in terms of the Feynman path integral picture [21], the parallel becomes even more compelling: Instantaneous values taken by the quantum system at intermediate times, which are integrated over, play the role of "virtual neurons" [3].

In any case the real time evolution of the two-qubit system can be treated as a neural network, because its evolution is a nonlinear function of the various adjustable parameters (weights) of the Hamiltonian. For as long as coherence can be maintained experimentally, it is a quantum neural network (QNN). Thus, if we can find values for the parameters such that the set of $\{inputs, outputs\}$ matches a measure of entanglement, we can use this setup as an experimental means of measuring the entanglement of any prepared state of the system, whether that state is analytically known or unknown. In this paper we find those parameters by training a simulation of the two-qubit system to a set of four input-output pairs. We then test the simulated net on a large number of additional states. The net is said to have generalized if the results on the testing set are correct and consistent. We show that this is so.

3 Dynamic Learning for the Quantum System

The goal of learning as applied to this quantum system is to control the system via the external parameters (tunneling, field and coupling values) to force it to calculate target outputs in response to given inputs. This is essentially a neural network supervised learning paradigm extended to the quantum system. The method, derived below, follows the methodology of Yann LeCun's Lagrangian formulation derivation of backpropagation [22] and Paul Werbos's description of backpropagation through time [23], and follows some of our earlier work [24, 25, 26] on learning in non-linear optical materials and in training of quantum Hopfield networks.

Our method uses the density matrix representation for generality.

We derive a learning rule for the quantum system based on dynamic backpropagation for time dependent recurrent neural networks. Given an input (initial density matrix), $\rho(0)$, and a target output, d, a training pair from a training set, we want to develop a weight update rule based on gradient descent to adjust the system parameters (tunneling, field and coupling), *i.e.*, train the system "weights", to reduce the squared error between the target, d, and the output, *Output*. While training the weights, the system's density matrix, $\rho(t)$, is constrained to satisfy the Schrdinger equation for all time in the interval $(0, t_f)$.

We define a Lagrangian, L, to be minimized as

$$L = \frac{1}{2}[d - \langle O(t_f)\rangle]^2 + \int_0^{t_f} \lambda^+(t)(\frac{\partial \rho}{\partial t} + \frac{i}{\hbar}[H, \rho])\gamma(t)dt \tag{3}$$

where the Lagrange multiplier vectors are $\lambda^+(t)$ and $\gamma(t)$ (row and column, respectively), and O is an output measure (or some function of a measure), which can be specified for the particular problem under consideration and is defined as:

$$Output = \langle O(t_f)\rangle = tr[\rho(t_f)O] = \sum_i p_i |\psi_i(t_f)\rangle\langle\psi_i(t_f)|O = \sum_i p_i \langle\psi_i(t_f)|O|\psi_i(t_f)\rangle \tag{4}$$

where tr stands for the trace of the matrix. We take the first variation of L with respect to ρ, set it equal to zero, then integrate by parts to give the following equation which can be used to calculate the vector elements of the Lagrange multipliers that will be used in the learning rule:

$$\gamma_i \frac{\partial \gamma_j}{\partial t} + \frac{\partial \lambda_i}{\partial t}\gamma_j - \frac{i}{\hbar}\sum_k \lambda_k H_{ki}\gamma_j + \frac{i}{\hbar}\sum_k \lambda_i H_{jk}\gamma_k = 0 \tag{5}$$

with the boundary conditions at the final time t_f given by

$$-[d - \langle O(t_f)\rangle]O_{ji} + \lambda_i(t_f)\gamma_j(t_f) = 0 \tag{6}$$

The gradient descent learning rule is given by

$$w_{new} = w_{old} - \eta \frac{\partial L}{\partial w} \tag{7}$$

for each weight parameter w, where η is the learning rate and

$$\frac{\partial L}{\partial w} = \frac{i}{\hbar}\int_0^{t_f} \lambda^+(t)[\frac{\partial H}{\partial w}, \rho]\gamma(t)dt = \frac{i}{\hbar}\int_0^{t_f} \sum_{ijk}(\lambda_i(t)\frac{\partial H_{ik}}{\partial w}\rho_{kj}\gamma_j - \lambda_i(t)\rho_{ik}\frac{\partial H_{kj}}{\partial w}\gamma_j)dt \tag{8}$$

Note that because of the Hermiticity of the Hamiltonian, H, and the density matrix ρ, $\lambda_i\gamma_j = \lambda_j\gamma_i$ and the derivative of the Lagrangian, L, with respect to the weight, w, as given by (8), will be a real number. This simplifies the calculation somewhat.

The time evolution of the quantum system is calculated by integrating the Schrdinger equation in MATLAB Simulink [27]. The ODE4 fixed step size solver was used with a step size of 0.05 ns. Discretization error for the numerical integration was checked by redoing the calculations with a timestep of a tenth the size; results were not affected. Since the error needs to be back propagated through time, the integration has to be carried out from t_f to 0.

To implement this in MATLAB Simulink, a change of variable is made by letting $t' = t_f - t$. Instead of using Simulink, the Schrdinger wave equation can also be integrated with any standard numerical method such as a FORTRAN or C-code program, which we have also done, to validate the MATLAB simulation.

4 Benchmark Training Results for Two-qubit Gates

We now train a two-SQUID quantum system in simulation, to produce two-input one-output classical logic gates, specifically the XOR and XNOR. The system is initialized to (prepared in) the input states shown in Table 1 and allowed to evolve for 300 ns. We choose as the output measure O the state of the second qubit, B, at the final time (*i.e.*, $O = \sigma_{zB}(t_f)$.). We call B the "target" qubit (while A is the "control" qubit.) The measure is applied to find the state of the system at this final time t_f and compared to the target output for that particular input. The parameters λ and γ are initialized with this error according to (6), and this is dynamically back propagated through time according to (5) and the weights updated according to (7). Since the control qubit A does not change its state and we are only concerned with measuring the output state of the target qubit B at the final time, we do not need to train the weights corresponding to the parameters for the control SQUID A, *i.e.*, K_A and ε_A need not be trained. Thus, we only train the weights corresponding to the parameter values K_B, ε_B and ζ_{AB}. The value for ε_A can be chosen to be any arbitrarily high value, say 1.0 GHz; this maintains the control qubit A in its original state. The initial values of the parameters (weights) before training are shown in Tables 2 and 3 for the XOR and XNOR gates, respectively, along with the final trained parameters (weights) for each logic gate. The field ε_B is allowed to vary with time. The simulation models this by allowing the field to vary every 100 ns, as a series of step functions: $\varepsilon_B(1)$ and $\varepsilon_B(2)$. The trained responses are shown in Table 1 along with the RMS error for each logic gate and the number of epochs of training used for each. The trained parameters agree with our previous analytic results[20]. Note that the parameter values can be rescaled overall.

Table 1. Training data for two input one output logic gates.

Input	XOR		XNOR	
$\|\psi\rangle$	Target	Output	Target	Output
$\|00\rangle$	-1	-0.9919	+1	0.9902
$\|01\rangle$	+1	0.9920	-1	-0.9903
$\|10\rangle$	+1	0.9902	-1	-0.9919
$\|11\rangle$	-1	-0.9903	+1	0.9920
RMS	0.00446		0.00447	
Epochs	300			

5 Quantum Control

It should be noted that our chosen measurement, $O = \sigma_{zB}(t_f)$, does not check the phase of the final state. That is, Tables 1-3 show only the acquisition of the *classical* gates XOR and XNOR. In order to show that we have a *quantum* gate we need to train the actual output

state, directly. That is, given an input state, we wish to adjust the parameters of the system such that it evolves to another given state. This is also known as "quantum control" [28]. Of course, we cannot use the density matrix approach if we wish to do this. A similar training procedure using kets is as follows. The output of the quantum system is taken as the overlap of the state of the system at the final time, t_f, with the desired state:

$$Out = \langle \psi_{desired} | \psi(t_f) \rangle \, . \tag{9}$$

We define a Lagrangian, L, to be minimized as

$$L = \frac{1}{2}|1 - \langle \psi_{desired} | \psi(t_f) \rangle|^2 + \int_0^{t_f} \lambda^* [\frac{d|\psi\rangle}{dt} - \frac{1}{i\hbar} H|\psi\rangle] dt + \int_0^{t_f} [\frac{d|\psi\rangle}{dt} - \frac{1}{i\hbar} H|\psi\rangle]^* \lambda dt \, , \tag{10}$$

where * indicates the complex conjugate. To minimize L, we set the first variation of L with respect to $|\psi\rangle$ equal to zero. After integration by parts, this gives a differential equation for λ as

$$\frac{d\lambda}{dt} = \frac{1}{i\hbar} H^* \lambda \, , \tag{11}$$

with a condition at the final time as

$$\lambda^*(t_f) = \mathrm{Re}[(1 - \langle \psi(t_f) | \psi_{des} \rangle) \langle \psi_{des}|] \, . \tag{12}$$

Again, we can solve this equation backward in time, and take the variation of L with respect to a weight parameter, w, where w represents any one of the parameters from the Hamiltonian, H, such as K_A, K_B, etc., to give the parameter (weight) update learning rule (7), where now

$$\frac{\partial L}{\partial w} = \int_0^{t_f} 2\mathrm{Re}[\frac{1}{i\hbar} \lambda^* \frac{\partial H}{\partial w} |\psi\rangle] dt \, , \tag{13}$$

and η is again the learning rate. Training using this method is less stable than with the density matrix; in particular, if we try to pin the control qubit A as before, with a large value of ε_A, there are rapid oscillations in the phase which make computation difficult. Fortunately one can get the same pinning result by setting $K_A = 0$. (Use of the density matrix method gets rid of this phase oscillation problem but of course does not allow the user to determine the phase of the output.)

Results are shown in Table 4. This is not the CNOT, since one of the qubits has an extra phase[11] of π; however, this gate, followed by the "controlled Z" gives the CNOT. (The

Table 2. Initial and trained parameters (weights) for XOR gate, in MHz.

Parameter (MHz)	Initial	XOR-trained
K_A	2.1333	2.1333
K_B	2.1333	1.2684
ζ_{AB}	0.1	-0.97981
$\varepsilon_A(1) = \varepsilon_A(2)$	1,000	1,000
$\varepsilon_B(2)$	0.1	1.0518
$\varepsilon_B(3)$	0.1	1.0534

solution for single-qubit phase shifts gates is obvious: set $\zeta_{AB} = 0$, and let ε_B be nonzero for the amount of time necessary.)

It should be noted that this, like the application in the previous section, is a "blind" application of our method. Recent work [12] especially by Khaneja, *et al.*, using a geometric approach, shows that careful analysis can produce a much more optimal (efficient) realization of a quantum operator. Still, our method allows relatively quick learning, and can be readily applied even when the problem -or operator –is not well understood, as with the entanglement witness in the next section.

6 Entanglement Witness

Of course it is known how to produce simple universal gates with any of a number of physical implementations, and as long as we know how to decompose our desired computation into those simpler blocks we do not need to do it "all at once." Showing that our quantum neural network can be trained to do progressively more complicated gates does have some advantages: Online training automatically adjusts for small effects not taken into account in whatever model was used to design the algorithm, for example.

However of much more interest is the possibility that a neural or AI approach can help us calculate things we do not have an algorithm for, and/or which we do not know how to decompose into simple gates. Entanglement is a good example. It is widely thought that the power of quantum computing and communications relies heavily on the use and manipulation of entanglement [5, 6]. But the quantification of entanglement is still not fully understood even for pure states, and for mixed states the situation is cloudier. For many practical applications we will also need an extension to systems of more than two qubits [14] or to entangled pairs of N-state systems, "quNits" [15].

Two prominent universal measures do exist: those of Bennett *et al.*[29] and Wootters[30], and of Vedral *et al.*[31]. The first is the entanglement of formation, which for pure states is equal to the von Neumann entropy of the reduced single-qubit states of the two-qubit system. The second constructs a space of density matrices, containing a subspace of unentangled states, and defines as the entanglement of a state the minimum distance of its density matrix to that subspace. The two measures do not give the same answer for a number of states, but each is internally consistent. We now use our training method to map an entanglement witness to a single (local) measure at the final time[32]. Unlike the method of Vedral, it does not require a minimization procedure, which can become cumbersome especially if the number of qubits is large; unlike both, it can in principle easily be used experimentally to

Table 3. Initial and trained parameters (weights) for XNOR gate, in MHz.

Parameter (MHz)	Initial	XNOR-trained
K_A	2.1333	2.1333
K_B	2.1333	1.2682
ζ_{AB}	0.1	0.97973
$\varepsilon_A(1) = \varepsilon_A(2)$	1,000	1,000
$\varepsilon_B(2)$	0.1	1.0508
$\varepsilon_B(3)$	0.1	1.0524

measure the entanglement of an unknown state. We show that our method approximately reproduces the entanglement of formation for large classes of states.

We return to a two-qubit system, for which the Hamiltonian is given by (1). We take as our output the square of the two-qubit correlation function at the final time, that is, $\langle O(t_f)\rangle = \langle \sigma_{zA}(t_f)\sigma_{zB}(t_f)\rangle^2 = [tr(\rho(t_f)\sigma_{zA}\sigma_{zB})]^2$. (We use here the square of the correlation function so that the range of the output will be [0,1] for convenience; the only modification necessary in Eq. (8) is the multiplication by $2\langle \sigma_{zA}(t_f)\sigma_{zB}(t_f)\rangle$.) In Table 5 we present a representative sample of the kinds of states whose entanglement we wish to calculate, along with the entanglement of each as calculated by the methods of references[29, 30, 31]. The Bell and EPR states are maximally entangled, and product and (completely) mixed states are minimally entangled. We chose a training set of four: one completely entangled state, one unentangled state, one classically correlated but unentangled state, and one partially entangled state. We used a time of evolution of 1000 $\hbar$, enough for the system to go through one complete oscillation. Results are presented in Table 6. The RMS error for the set, after training, is essentially zero. Table 7 lists the parameter values the system trained to in order to achieve the entanglement witness. We then tested the witness on a testing set which included a maximally entangled state of a type not seen before by the net, the EPR state; a pure unentangled state; a correlated unentangled state, a different partially entangled state, and a mixed state. Results for the testing set are shown in Table 8. The error for the testing set is also essentially zero. A large number of permutations of possible states have been tried with exactly similar results. The quantum neural net has learned to compute an approximate general measure of entanglement.

The training for the partially entangled state P deserves some further comment. We noted, above, that there is no general agreement on what the entanglement of P is, though it ought to lie somewhere between 0 and 1 on the scale we are using. Therefore we trained the network for various different target values for the entanglement of P, including the numbers (0.32, 0.46, 0.55) calculated by the three comparison methods shown in Table 5. In Figure 1 we show the total error of the QNN for both the training and testing sets, as a function of the desired value. There is a minimum at approximately 0.44317 (though that degree of precision is probably imaginary.) What this means is that using the value of 0.44317 for the fourth training pair significantly increased the compatibility of the set of training and testing pairs, taken together; while we could train the state P to any value we desire, only if we set that value to 0.44317 can we get at the same time the values we wish for the other training and testing pairs. We can say that in some sense the value of about 0.44 is the

Table 4. Training of the CNOT.

Input	Output amplitudes			
$\|\psi\rangle$	$\|00\rangle$	$\|01\rangle$	$\|10\rangle$	$\|11\rangle$
$\|00\rangle$	$0.9998 + 0.009i$	$0.0006 + 0.0188i$	0.0000	0.0000
$\|01\rangle$	$0.0006 + 0.0188i$	$0.9977 - 0.0653i$	0.0000	0.0000
$\|10\rangle$	0.0000	0.0000	$0.0323 + 0.0i$	$-0.9993 + 0.0202i$
$\|11\rangle$	0.0000	0.0000	$0.9993 + 0.0203i$	$0.0292 - 0.0085i$
$K_A = 0.0 = \varepsilon_A(1) = \varepsilon_A(2)$; $\zeta_{AB} = 0.0096$				
$K_B = 0.0054$; $\varepsilon_B = (-0.0606, -0.0133, -0.0055)$				

Table 5. Some possible states of the two-qubit system. The relative amplitudes (for the ket states) are given without normalization for clarity. The first two are maximally entangled. The second two are product states (flat = $(|0\rangle + |1\rangle)_A(|0\rangle + |1\rangle)_B$ and $C = |0\rangle_A(|0\rangle + \gamma|1\rangle)_B$) and thus have zero entanglement; and P is partially entangled. M is a mixed state and cannot be expressed as a ket; its density matrix is given instead. The classical correlation is computed as $\langle\sigma_{zA}(0)\sigma_{zB}(0)\rangle$. C and M are classically correlated but not entangled. The last three columns show the entanglement as calculated by the methods of Bennett[29] and Vedral [31], using the von Neumann metric and the Bures metric. Both distance measures have been normalized to unity.

State	Relative amplitudes of				Classical	Entanglement		
	$\|00\rangle$	$\|01\rangle$	$\|10\rangle$	$\|11\rangle$	Correlation	Bennett	von Neumann	Bures
Bell	1	0	0	$e^{i\theta}$	1	1	1	1
EPR	0	1	$e^{i\theta}$	0	-1	1	1	1
flat	1	1	1	1	0	0	0	0
C	1	γ	0	0	$\frac{1-\|\gamma\|^2}{1+\|\gamma\|^2}$	0	0	0
P	0	1	1	1	$-1/3$	0.55	0.32	0.46
M	$(\|00\rangle\langle 00\| + \|11\rangle\langle 11\|)/2$				1	0	0	0

Table 6. Training data for QNN entanglement witness.

Input state	Initial	Desired	Trained
Bell, $\delta = 0$	1.0	1.0	0.99997
Flat	0.0	0.0	2.01×10^{-6}
C, $\gamma = 0.5$	0.36	0.0	2.61×10^{-5}
P	0.11	0.44317	0.44317
RMS		1.08×10^{-5}	
Epochs		2000	

Table 7. Initital and trained parameters for entanglement, in MHz.

Parameter (MHz)	Initial	Trained
$K_A(1)$	2.5	2.3576
$K_A(2)$	2.5	2.3576
$K_A(3)$	2.5	2.3577
$K_A(4)$	2.5	2.3461
$K_B(1)$	2.5	2.3576
$K_B(2)$	2.5	2.3576
$K_B(3)$	2.5	2.3576
$K_B(4)$	2.5	2.3546
$\zeta(1)$	0.1	0.045026
$\zeta(2)$	0.1	0.10117
$\zeta(3)$	0.1	0.10771
$\zeta(4)$	0.1	0.044221
$\varepsilon_A(1)$	0.1	0.10913
$\varepsilon_A(2)$	0.1	0.03768
$\varepsilon_A(3)$	0.1	0.08671
$\varepsilon_A(4)$	0.1	0.071464
$\varepsilon_B(1)$	0.1	0.10913
$\varepsilon_B(2)$	0.1	0.063774
$\varepsilon_B(3)$	0.1	0.038802
$\varepsilon_B(4)$	0.1	0.072387

Table 8. Representative testing results for the quantum neural network. P_2 is the state $(|00\rangle + |10\rangle + |11\rangle)/\sqrt{3}$. Parameters used are listed in Table 7.

State	Desired	QNN Output
EPR, $\theta = 0$	1.0	1.0
EPR, $\theta = \pi$	1.0	1.0
Bell, $\delta = \pi$	1.0	0.99997
$\|00\rangle$	0.0	3.24×10^{-5}
$\|10\rangle + 0.9\|11\rangle$	0.0	3.39×10^{-6}
P_2	0.44317	0.44317
M	0.0	2.59×10^{-13}
RMS	7.68×10^{-6}	

"natural" value for the entanglement of this state, at least as computed by this net; that is, using this value as the target value for P leads to the greatest possible self-consistency for the method. Interestingly 4/9= 0.44444 is the value for the entanglement of state P as calculated by the formula $tr(\rho\tilde{\rho})$, where $\tilde{\rho} = (\sigma_y \otimes \sigma_y)\rho^*(\sigma_y \otimes \sigma_y)$, which for pure states is a monotonically increasing function (like the concurrence) and thus could be used as a possible measure of entanglement. This measure, however, fails for mixed states: in particular, it gives an entanglement of 1 for the mixed state M (Table 5). (Coincidentally the number 0.44229 for the entanglement of formation also has some special significance [33] for partially mixed states: this is the point at which the Werner states[34] begin to violate the Bell inequality.)

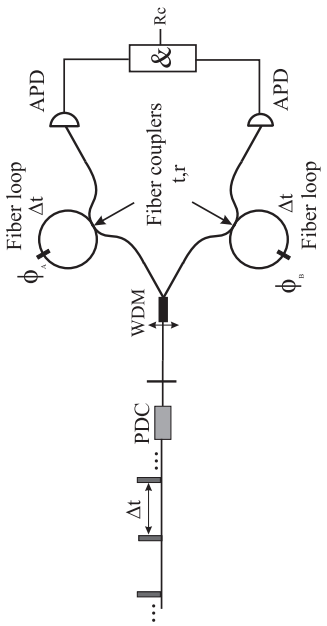

Fig. 1. Total error, including training and testing, for different target values for the partially entangled state P. In each run, represented by a data point on the graph, all four of the training set pairs were trained, and only the desired value for the state P was changed; all were tested on the same testing set as shown in Table 8. The minimum error found, 6.27×10^{-6}, occurs at a target value of 0.44317.

It may also be of interest to note that almost all the contribution to the error comes from the partially entangled states in the training and testing sets. So, for example, the total error (from training and testing sets) for the target value of 0.32 for P is about 0.2; most of this comes from the testing of state P_2, for which the net calculates 0.41942 instead of 0.32. That is, except at the target value of about 0.44, the net does not recognize P_2 as being "the same" as P. Results for the mixed states are almost exactly the same irrespective of the target value for P: if they were plotted in Figures 3 and 4, they would lie almost on top of the QNN values shown. The results for completely mixed states are always calculated to be zero (10^{-6} or smaller.)

Once the net is trained it is then a simple matter to calculate the output value (square of the correlation function) which would be measured at the given final time for any state at all, pure or mixed. In addition to the testing set, we set up a grid to check that the calculated entanglement of all pure product states is zero, that is, all states of the form $(\alpha|1\rangle + \beta|0\rangle)_A(\gamma|1\rangle + \delta|0\rangle)_B/\sqrt{\alpha\gamma + \alpha\delta + \beta\gamma + \beta\delta}$. The RMS error for the set of 10,000 was 1.1×10^{-7}. Similarly we tested a grid for mixed states; the RMS error for the set of 10,000 was 2.6×10^{-7}. Thus for both pure and mixed separable states the QNN gives reasonably good results.

We compare the QNN method with both a (widely-accepted) measure (Bennett and Wootters's entanglement of formation) and a witness (Toth and Guhne's local entanglement witness [35], W_{GHZn}, which for the two-qubit system considered here is equal to $I - \sigma_{xA}\sigma_{xB} - \sigma_{zA}\sigma_{zB}$.) This is because, while we claim to have designed only a witness, not a measure, our method turns out to be somewhat better than expected: to some extent, it gives information, as well, on the amount of entanglement present. When we compare the QNN to the entanglement of formation, we are looking for numerical agreement and get it to some extent; when we compare both Toth and Guhne's W and the QNN, as witnesses, to the entanglement of formation, we look, as is appropriate, only for agreement as to whether or not entanglement is present. Note that this last measures proximity to the Bell triplet state and that a negative number indicates entanglement.

Figure 2 shows the QNN entanglement for the pure state $P_3(\gamma) = \frac{|00\rangle + |11\rangle + \gamma|01\rangle}{\sqrt{2+|\gamma|^2}}$, as a function of γ. Values for the entanglement of formation are also shown for comparison; the results are very similar and constitute good agreement. The W witness correctly indicates the presence of entanglement until the contamination gets too large: for $\gamma = 1$ W is zero, which indicates no entanglement and is incorrect. Thus here the QNN does better as a witness than W. The disagreement is probably because the W witness is only good for states close to the Bell triplet state $|\Phi^+\rangle$; for $\gamma = 1$ we are probably sufficiently far from the Bell triplet state that the witness no longer applies well.

Figure 3 shows the calculated entanglement for the Bell triplet Werner[34] mixed states as a function of fidelity, again, with Bennett and Wootters's entanglement of formation and Toth and Guhne's witness W for comparison. Again, agreement of the QNN with the former is quite good though not exact. For fidelity between approximately 0.28 and 0.5 our method gives a small but nonzero entanglement, which is incorrect, since it has been shown[29] that for $0.25 < F < 0.5$ the Werner state can be written as a mixture of product states. Here W is a better witness than the QNN. In Figure 4 we show the QNN entanglement for the states $M'(\gamma) = \frac{\gamma|11\rangle\langle 11| + |\Phi^+\rangle\langle\Phi^+|}{\gamma+1}$, and compare those results to those for the entanglement

of formation. Again the agreement is good though not exact. The W witness here fails for $\gamma > 0.5$.

Using a larger training set -one that includes mixed states -does reduce the error shown in Figure 3, though with this model (as Figure 1 demonstrates) we have already minimized the error shown in Figure 2 by optimizing the target value for P. Changing the target value for P does not alter the results for Figure 3 significantly. We thought it more interesting to show results for what seems to be the minimum training set for an (approximate) entanglement witness.

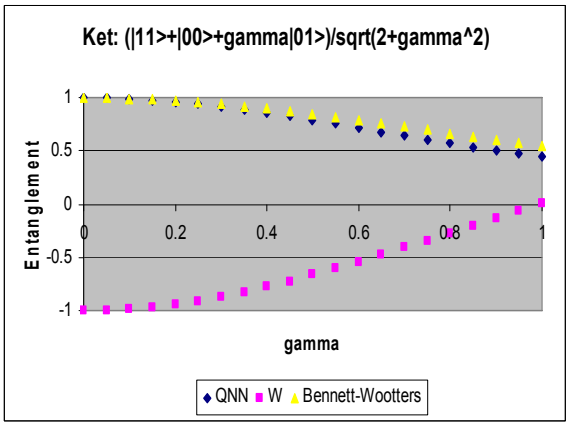

Fig. 2. Entanglement of the pure state $P_3(\gamma) = \frac{|00\rangle+|11\rangle+\gamma|01\rangle}{\sqrt{2+|\gamma|^2}}$, as a function of γ, as calculated by the QNN, using the trained parameters listed in Table 7, by Bennett's entanglement of formation [29, 30]; and by Toth and Guhne's local entanglement witness W [35]. Note that $W < 0$ indicates entanglement.

7 Discussion

It will have been noticed that all the coefficients on states so far trained or tested were real. It is a natural question at this point to ask about a phase difference, *e.g.*, between the two parts of a Bell state. In Figure 5 we show the calculated correlation function for the Bell state $|00\rangle + e^{i\theta}|11\rangle$ as a function of initial phase difference θ. (The EPR state shows exactly the same dependence.) Of course the actual entanglement is not a function of the phase difference, so here the QNN measure is wrong; or rather, this is the major reason our method produces only a "witness" not a "measure." Interestingly Toth-Guhne's W witness shows a similar oscillation, though at half the frequency.

Product states show a similar if smaller amplitude oscillation. Figure 6 shows that the oscillation in the product state C is so small as to be negligible; W correctly predicts no entanglement though we are surely too far from the Bell state for the method to be applicable. Figure 7 shows the oscillation for P_2 when the phase difference is between the entangled pair and $|01\rangle$. The "correct" target value for the P states depends on the phase difference, and is equal to 0.44317 only for $\theta = 0$; the mean value is very close to 4/9 (0.44444 to five significant figures, for twenty evaluated points between 0 and 2π.) Figure 8 shows the oscillation for

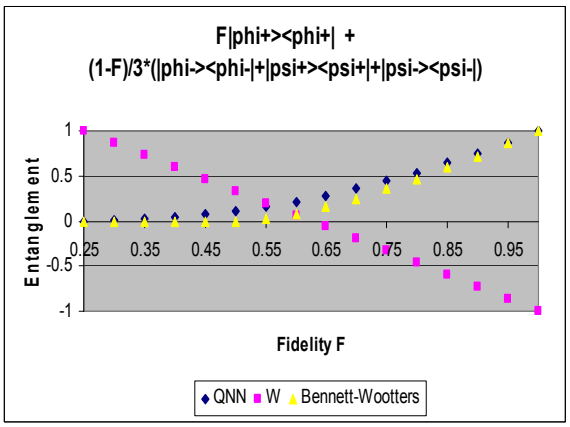

Fig. 3. Entanglement for the Bell triplet Werner mixed states $F|\Phi^+\rangle\langle\Phi^+|+(1-F)/3)(|\Psi^+\rangle\langle\Psi^+|+|\Psi^-\rangle\langle\Psi^-|+|\Phi^-\rangle\langle\Psi^-|)$ as a function of fidelity F, where $|\Psi^\pm\rangle = (|\uparrow\downarrow\rangle \pm |\downarrow\uparrow\rangle)/\sqrt{2}$, and $|\Phi^\pm\rangle = (|\uparrow\uparrow\rangle \pm |\downarrow\downarrow\rangle)/\sqrt{2}$, as calculated by Bennett/Wootters [29, 30], by QNN, and by Toth/Guhne's W witness[35]. Again the QNN parameters were as listed in Table 7. The Werner states are $x = (4F-1)/3$ parts pure triplet (fully entangled), and $(1-x)$ parts identity operator (completely mixed)[29]. Note that $W < 0$ indicates entanglement.

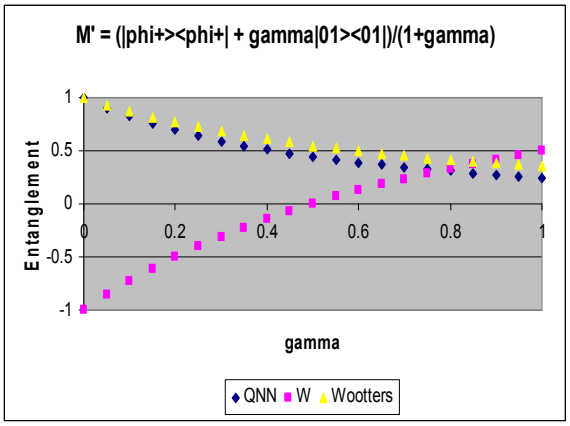

Fig. 4. Entanglement for the states $M'(\gamma) = (\gamma|01\rangle\langle 01| + |\Phi^+\rangle\langle\Phi^+|)/(1+\gamma)$, calculated by entanglement of formation, by QNN, and by Toth/Guhne's W. For $\gamma = 1$ this is Bennett's M state; the QNN calculates its entanglement as 0.24716, while Wootters's method[30] gives 0.35458. QNN parameters were as listed in Table 7. Note that $W < 0$ indicates entanglement.

P_2 when the phase difference is within the entangled pair. The oscillation from Figure 5 is reproduced, as expected, but on a smaller amplitude scale (0 to 0.44 rather than 0 to 1) because of the presence of the amplitude in the $|01\rangle$ state. In this case, as in Figure 5, we can see the oscillation in the W witness as well. Note that for this state, for any value of θ, the W witness predicts no entanglement; this is doubtless due to its being insufficiently "close"to the Bell state. The best point is at $\theta = 0$ (or 2π); this is the same as the point at $\gamma = 1$ in Figure 2.

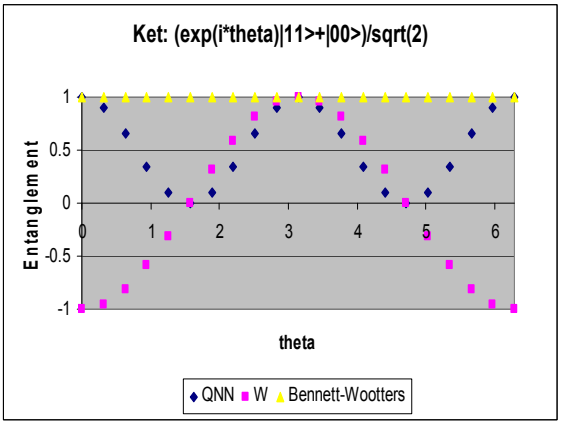

Fig. 5. Calculated entanglement for the state $|00\rangle + e^{i\theta}|11\rangle$, as a function of θ, by Bennett-Wootters, QNN, and Toth-Guhne's W. QNN parameters were as listed in Table 7. Note that $W < 0$ indicates entanglement.

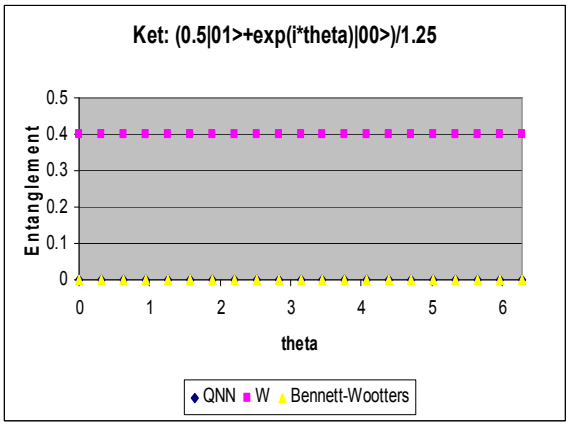

Fig. 6. Calculated entanglement for the C state, by Bennett-Wootters, QNN, and Toth-Guhne's W. QNN parameters were as listed in Table 7. Note that $W < 0$ indicates entanglement.

We can see why both witnesses behave this way by considering that any single measurement must be of the form of the trace of the initial density matrix times some Hermitian operator.

Let that matrix be given by $\begin{pmatrix} e_1 & a_1+ib_1 & a_2+ib_2 & a_3+ib_3 \\ a_1-ib_1 & e_2 & c_1+id_1 & c_2+id_2 \\ a_2-ib_2 & c_1-id_1 & e_3 & c_3+id_3 \\ a_3-ib_3 & c_2-id_2 & c_3-id_3 & e_4 \end{pmatrix}$ Now, for the case of the initial state's being $|00\rangle + e^{i\theta}|11\rangle$, this means that the expectation value is given by $e_1 + e_4 + 2\mathrm{Re}[e^{i\theta}(a_3 - ib_3)]$. Unless both a_3 and b_3 are zero this necessarily depends on θ; even if we set them both to zero we still have to deal with, *e.g.*, the case of the state $|01\rangle + e^{i\theta}|10\rangle$. Thus it is not possible to design a single measurement as a completely general entanglement measure. The oscillation that is seen in both our witness and that of Toth and Guhne is inescapable. Thus in order to use our method to measure independently the entanglement of an unknown state, it is necessary to do at least one other measurement and perhaps two. Recently Yang and Han[36] have devised a means of extracting an arbitrary relative phase from a multiqubit entangled state by local Hadamard transformations and measurements along a single basis; this method together with our own, then, can be used as an unambiguous entanglement witness.

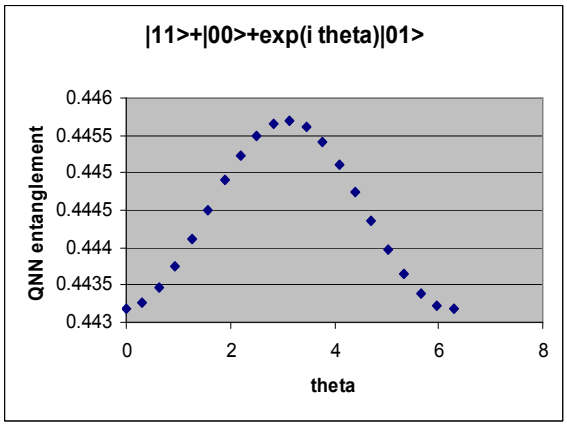

Fig. 7. Entanglement of the state $|00\rangle + |11\rangle + e^{i\theta}|01\rangle$, , as a function of θ, as calculated by QNN. QNN parameters were as listed in Table 7. Toth-Guhne's W is zero for all values of θ (too far from the Bell state; see Figure 2.)

One possible problem with the proposed method is that the output function chosen is always greater than or equal to zero; thus, in actual experimental measurements, it will systematically overestimate the entanglement for states close to separability. It should also be noted that the correlation function cannot be determined in a single measurement, since it is an average quantity. To measure the correlation function experimentally, even if the average should be very close to zero or to one, it would be necessary to produce the desired state many times; since in a given situation it may not be possible to do so, or without varying the phase difference, this may defeat the purpose. Nevertheless we believe this approach may be a fruitful one: it is certainly easier to measure the correlation function than it is to determine the density matrix in full, as standard calculational approaches require, or even the four parameters necessary to find the concurrence[37]. It is possible that a different, more clever choice of measurement operator(s) could reduce the number of measurements necessary still

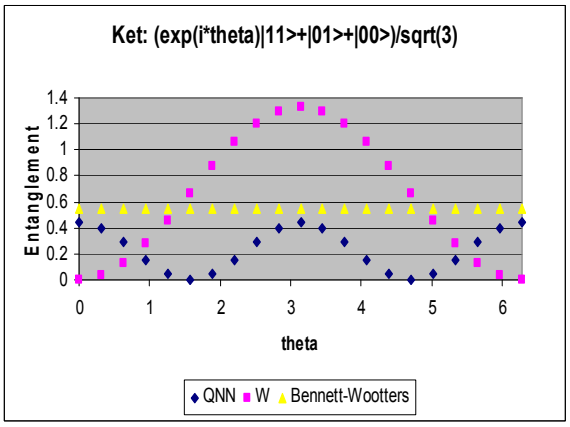

Fig. 8. Entanglement of the state $|00\rangle + e^{i\theta}|11\rangle + |01\rangle$, as a function of θ, as calculated by Bennett-Wootters, QNN, and Toth-Guhne's W. QNN parameters were as listed in Table 7. Note that $W < 0$ indicates entanglement.

further. In addition, generalization to multiple qubits or to quNits is straightforward: as long as a training set of sufficient size is used there is no reason to think that the net would be unable to learn the generalized measure. Our experience here seems to show that the necessary size is not large. We are currently pursuing these lines of research.

8 Conclusions

In this paper we have developed a general dynamic learning algorithm for training a quantum computer, for either pure or mixed states. We have demonstrated successful learning of some simple benchmark applications. We have also shown that this method can be used for the learning of an entanglement witness for an input state. We have shown that our witness approximately reproduces the entanglement of formation for large classes of states, and, while it suffers from a systematic oscillation problem, so must every single-measurement entanglement witness, and a method like Yang and Han's can be used in conjunction to take care of the problem. It is superior to other witnesses in that, first, the state need not be "close" to a particular kind of entangled state or, indeed, to any particular state; and, second, that the state may be completely unknown. Generalization to systems of more than two qubits and to multiple level systems is in progress.

Acknowledgements

This work was supported by the National Science Foundation, Grants ECS-9820606 and 0201995. We thank W.K. Wootters, W.J. Axmann, and A.J. Schauf for helpful discussions.

References

1. M. A. Nielsen and I. L. Chuang (2001), *Quantum Computation and Quantum Information*, Cambridge University Press (Cambridge, England).

2. G.P. Berman, G.D. Doolen, R. Mainieri and V.I. Tsifrinovich (1998), *Introduction to Quantum Computers*, World Scientific Publishing Co. (Singapore).
3. E. C. Behrman, L. R. Nash, J. E. Steck, V. G. Chandrashekar, and S. R. Skinner (2000), *Simulations of quantum neural networks*, Information Sciences 128, pp. 257-269.
4. C. H. Bennett, G. Brassard and N. D. Mermin (1992), *Quantum cryptography without Bell's theorem*, Phys. Rev. Lett. 68, pp. 557-559.
5. D. P. DiVincenzo (1995), *Quantum computation*, Science 270, pp. 255-261.
6. N. Linden, S. Popescu, and A. Sudbery(1999),*Nonlocal parameters for multiparticle density matrices*, Phys. Rev. Lett. 83, pp. 243 -247.
7. P. Shor (1997),*Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer*, SIAM J. Computing 26, pp. 1484-1509.
8. L. Grover (1996), *A fast quantum mechanical algorithm for database search*, in *Proc. 28th Annu. ACM Symposium on the Theory of Computing*, ACM (New York), pp. 212-219.
9. D. Aharonov, V. Jones, and Z. Landau (2006), *A Polynomial Quantum Algorithm for Approximating the Jones Polynomial*, quant-ph/0511096; S. Lomonaco, Jr., and L.H. Kauffman (2006), *Topological quantum computing and the Jones polynomial* , quant-ph/0605004.
10. M.A. Nielsen, M.R. Dowling, M. Gu, and A.C. Doherty (2006), *Quantum computation as geometry* Science 311, pp. 1133-1135; M.A. Nielsen (2005), *A geometric approach to quantum circuit lower bounds*, quant-ph/0502070.
11. D.P. DiVincenzo and J. Smolin (1994), *Results on two-bit gate design for quantum computers*, in *Proceedings of the Workshop on Physics and Computation (PhysComp '94)*, IEEE (Dallas, TX), pp.14-19.
12. N. Khaneja, B. Heitman, A. Sporl, H. Yuan, T. Schultebruggen, and S.J. Glaser (2006), *Quantum gate design metric*, quant-ph/0605071 v1 (2006); T. Schulte-Herbruggen, A. Sporl, N. Khaneja, and S.J. Glaser (2005), *Optimal control-based efficient synthesis of building blocks of quantum algorithms seen in perspective from network complexity towards time complexity*, quant-ph/0502104; T. Schulte-Herbruggen, A. Sporl, N. Khaneja, and S.J. Glaser (2006), *Optimal control for generating quantum gates in open dissipative systems*, quant-ph/0609037.
13. R. Filip (2002), *Overlap and entanglement-witness measurements* , Phys. Rev. A 65, 062320; F.G.S.L. Brando (2005), *Quantifying entanglement with witness operators* , quant-ph/0503152.
14. D.M. Greenberger, M.A. Horne, and A. Zeilinger (1989), in *Bell's Theorem and the Conception of the Universe*, M. Kafatos, ed. , Kluwer Acdemic (Dordrecht), p 107.
15. D. Kaszlikowski, P. Gnacinski, M. Zukowski, W. Miklaszewski, and A. Zeilinger (2000), *Violations of local realism by two entangled N-dimensional systems are stronger than for two qubits*, Phys. Rev. Lett. 85, pp. 4418-4421.
16. T. Yamamoto, Yu.A. Pashkin, O. Astafiev, Y. Nakamura, and J.S. Tsai (2003), *Demonstration of conditional gate operation using superconducting charge qubits*, Nature 425, pp. 941-944.
17. S. Gulde, M. Riebe, G. Lancaster, C. Becher (2003), *Implementation of the Deutsch-Jozsa algorithm on an ion-trap quantum computer*, Nature 421, pp. 48-50.
18. L.M.K. Vandersypen, M. Steffen, G. Breyta, CS Yannoni, MH Sherwood, and IL Chuang (2001), *Experimental Realization of Shor's Quantum Factoring Algorithm Using Nuclear Magnetic Resonance*, Nature 414, pp. 883-887 (2001).
19. C-P Yang and S. Han (2003), *An arbitrary-operation gate with a SQUID qubit*, quant-ph/0305004; Z. Zhou, S-I Chu, and S. Han (2002), *Quantum computing with superconducting devices: three-level SQUID qubit*, Phys Rev B 66, 054527 (2002); C-P Yang, S-I Chu, and S. Han (2003), *Possible realization of entanglement, logical gates, and quantum-information transfer with superconducting-quantum-interference-device qubits in cavity QED*, Phys Rev A 67, 042311.
20. P. K. Gagnebin, S.R. Skinner, E.C. Behrman, J.E. Steck, Z. Zhou, and S. Han (2005), *Quantum gates using a pulsed bias scheme*, Phys. Rev. A. 72, 042311.
21. R.P. Feynman (1951), *An operator calculus having applications in quantum electrodynamics*, Phys. Rev. 84, 108-128; R.P. Feynman and A.R. Hibbs (1965), *Quantum Mechanics and Path Integrals*, McGraw-Hill (New York).

22. Yann le Cun (1988), *A theoretical framework for back-propagation* in *Proc. 1998 Connectionist Models Summer School,* D. Touretzky, G. Hinton, and T. Sejnowski, eds., Morgan Kaufmann, (San Mateo), pp. 21-28.
23. Paul Werbos (1992), in *Handbook of Intelligent Control* , Van Nostrand Reinhold, p. 79.
24. J. E. Steck, S. R. Skinner, A. A. Cruz-Cabrera, M. Yang and E. C. Behrman (2000), *Field computation for artificial network hardware: Examples in non-linear optical materials*, in *4th International Conf. on Computational Intelligence and Neurosciences, JCIS 2000*, pp 790-794.
25. E. C. Behrman, J. E. Steck and S. R. Skinner (1999), *A spatial quantum neural computer*, in *International Joint Conf. on Neural Networks (IJCNN99)*, IEEE (Washington, DC), pp. 874-877.
26. S. R. Skinner, E. C. Behrman, A. A. Cruz-Cabrera, and J. E. Steck (1995) *Neural network implementation using self-lensing media*, Applied Optics 34, pp. 4129-4135.
27. MATLAB Simulink documentation notes [Online]. Available at http://www.mathworks.com/access/helpdesk/help/toolbox/simulink
28. M Shapiro and P. Brumer (2003), *Principles of the Quantum Control of Molecular Processes*, Wiley (Hoboken, NJ).
29. C.H. Bennett, D.P. DiVincenzo, J.A. Smolin, and W.K. Wootters (1996), *Mixed-state entanglement and quantum error correction*, Phys. Rev. A 54, pp. 3824-3851.
30. W.K. Wootters (1998), *Entanglement of formation of an arbitrary state of two qubits* , Phys. Rev. Lett. 80, pp. 2245-2248.
31. V. Vedral, M.B. Plenio, M.A. Rippin, and P.L. Knight (1997), *Quantifying entanglement*, Phys. Rev. Lett. 78, pp. 2275-2279; V Vedral and M.B. Plenio (1998), *Entanglement measures and purification procedures*, Phys. Rev. A 57, pp. 1619-1633; L. Henderson and V. Vedral (2001), *Classical, quantum and total correlations*, J. Phys. A 34, pp. 6899-6905.
32. Other workers have proposed entanglement detection through dynamics + local measurement. See, *e.g.*, R. Rahimi, A. SaiToh, M. Nakahara, and M. Kitagawa (2007), *Single-experiment-detectable multipartite entanglement witness for ensemble quantum computing* , quant-ph/0608168; G. Toth (2004), *Entanglement detection in optical lattices of bosonic atoms with collective measurements*, Phys. Rev.A 69, 052327.
33. W.J. Munro, K. Nemoto and A.G. White (2001), *Maximally entangled mixed states and the Bell inequality* J. Mod. Optics 48, pp. 1239-1246.
34. R.F. Werner (1989), *Quantum states with Einstein-Podolsky-Rosen correlations admitting a hidden-variable model*, Phys. Rev. A 40, pp. 4277-4281.
35. G. Toth and O. Guhne (2005), *Detecting genuine multipartite entanglement with two local measurements*, Phys. Rev. Lett. 94, 060501.
36. C-P Yang and S. Han (2005), *Extracting an arbitrary relative phase from a multiqubit two-component entangled state*, Phys. Rev. A 72, 014306.
37. P. Horodecki (2001), *Measuring quantum entanglement without prior state reconstruction*, quant-ph/0111082

Quantum Information and Computation, Vol. 0, No. 0 (2003) 000–000

INSTRUCTIONS FOR TYPESETTING AN ARTICLE FOR QUANTUM INFORMATION AND COMPUTATION[a]

FIRST AUTHOR[b]
University Department, University Name, Address
City, State ZIP/Zone, Country[c]

SECOND AUTHOR
Group, Laboratory, Address
City, State ZIP/Zone, Country

Received (received date)
Revised (revised date)

Your abstract goes here.

Keywords: The contents of the keywords
Communicated by: to be filled by the Editorial

1 Introduction

The journal of *Quantum Information and Computation*, for both on-line and in-print editions, will be produced by using the latex files of manuscripts provided by the authors. It is therefore essential that the manuscript be in its final form, and in the format designed for the journal because there will be no futher editing. The authors are strongly encouraged to use Rinton latex template to prepare their manuscript. Or, the authors should please follow the instructions given here if they prefer to use other software. In the latter case, the authors ought to provide a postscript file of their paper for publication.

2 Text

Contributions are to be in English. Authors are encouraged to have their contribution checked for grammar. Abbreviations are allowed but should be spelt out in full when first used.

The text is to be typeset in 10 pt Times Roman, single spaced with baselineskip of 13 pt. Text area (excluding running title) is 5.6 inches across and 8.0 inches deep. Final pagination and insertion of running titles will be done by the editorial. Number each page of the manuscript lightly at the bottom with a blue pencil. Reading copies of the paper can be numbered using any legible means (typewritten or handwritten).

[a]Typeset the title in 10 pt Times Roman, uppercase and boldface.
[b]Typeset names in 10 pt Times Roman, uppercase. Use the footnote to indicate the present or permanent address of the author.
[c]State completely without abbreviations, the affiliation and mailing address, including country. Typeset in 8 pt Times Italic.

3 Headings

Major headings should be typeset in boldface with the first letter of important words capitalized.

3.1 Sub-headings

Sub-headings should be typeset in boldface italic and capitalize the first letter of the first word only. Section number to be in boldface roman.

3.1.1 Sub-subheadings

Typeset sub-subheadings in medium face italic and capitalize the first letter of the first word only. Section number to be in roman.

3.2 Numbering and Spacing

Sections, sub-sections and sub-subsections are numbered in Arabic. Use double spacing before all section headings, and single spacing after section headings. Flush left all paragraphs that follow after section headings.

3.3 Lists of items

Lists may be laid out with each item marked by a dot:

- item one,
- item two.

Items may also be numbered in lowercase roman numerals:

(i) item one
(ii) item two
   (a) Lists within lists can be numbered with lowercase roman letters,
   (b) second item.

4 Equations

Displayed equations should be numbered consecutively in each section, with the number set flush right and enclosed in parentheses.

$$\mu(n,t) = \frac{\sum_{i=1}^{\infty} 1(d_i < t, N(d_i) = n)}{\int_{\sigma=0}^{t} 1(N(\sigma) = n) d\sigma} \,. \tag{1}$$

Equations should be referred to in abbreviated form, e.g. "Eq. (1)" or "(2)". In multiple-line equations, the number should be given on the last line.

Displayed equations are to be centered on the page width. Standard English letters like x are to appear as x (italicized) in the text if they are used as mathematical symbols. Punctuation marks are used at the end of equations as if they appeared directly in the text.

Theorem 1: Theorems, lemmas, etc. are to be numbered consecutively in the paper. Use double spacing before and after theorems, lemmas, etc.

Proof: Proofs should end with □.

5 Illustrations and Photographs

Figures are to be inserted in the text nearest their first reference. The postscript files of figures can be imported by using the commends used in the examples here.

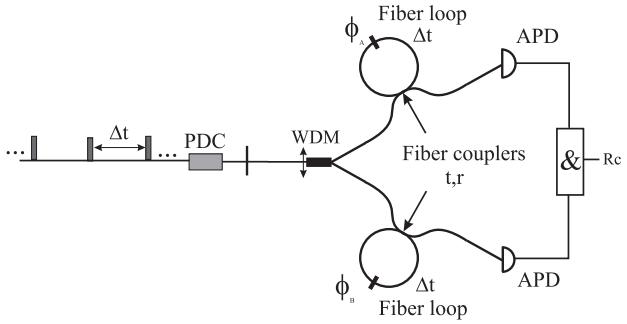

Fig. 1. figure caption goes here.

Figures are to be sequentially numbered in Arabic numerals. The caption must be placed below the figure. Typeset in 8 pt Times Roman with baselineskip of 10 pt. Use double spacing between a caption and the text that follows immediately.

Previously published material must be accompanied by written permission from the author and publisher.

6 Tables

Tables should be inserted in the text as close to the point of reference as possible. Some space should be left above and below the table.

Tables should be numbered sequentially in the text in Arabic numerals. Captions are to be centralized above the tables. Typeset tables and captions in 8 pt Times Roman with baselineskip of 10 pt.

Table 1. Number of tests for WFF triple NA = 5, or NA = 8.

		NP			
		3	4	8	10
NC	3	1200	2000	2500	3000
	5	2000	2200	2700	3400
	8	2500	2700	16000	22000
	10	3000	3400	22000	28000

If tables need to extend over to a second page, the continuation of the table should be preceded by a caption, e.g. "(*Table 2. Continued*)."

7 References Cross-citation

References cross-cited in the text are to be numbered consecutively in Arabic numerals, in the order of first appearance. They are to be typed in brackets such as [1] and [2, 3, 4].

8 Sections Cross-citation

Sections and subsctions can be cross-cited in the text by using the latex command shown here. In Section 8, we discuss ....

9 Footnotes

Footnotes should be numbered sequentially in superscript lowercase Roman letters.[a]

Acknowledgements

We would thank ...

References

References are to be listed in the order cited in the text. For each cited work, include all the authors' names, year of the work, title, place where the work appears. Use the style shown in the following examples. For journal names, use the standard abbreviations. Typeset references in 9 pt Times Roman.

1. P. Horodecki and R. Horodecki (2001), *Distillation and bound entanglement*, Quantum Inf. Comput., Vol.1, pp. 045-075.
2. R. Calderbank and P. Shor (1996), *Good quantum error correcting codes exist*, Phys. Rev. A, 54, pp. 1098-1106.
3. M.A. Nielsen and J. Kempe (2001), *Separable states are more disordered globally than locally*, quant-ph/0105090.
4. A.W. Marshall and I. Olkin (1979), *Inequalities: theory of majorization and its applications*, Academic Press (New York).

Appendix A

Appendices should be used only when absolutely necessary. They should come after the References. If there is more than one appendix, number them alphabetically. Number displayed equations occurring in the Appendix in this way, e.g. (A.1), (A.2), etc.

$$\langle \hat{O} \rangle = \int \psi^*(x) O(x) \psi(x) d^3x \ . \tag{A.1}$$

[a]Footnotes should be typeset in 8 pt Times Roman at the bottom of the page.